



# UNITED STATES PATENT AND TRADEMARK OFFICE

3  
UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/477,688	01/06/2000	STEPHEN ANTHONY EDWARDS	1000/5	9589
35795	7590	10/04/2004	EXAMINER	
JONATHAN T. KAPLAN ATTORNEY AT LAW 140 NASSAU STREET NEW YORK, NY 10038-1501			ALI, SYED J	
		ART UNIT	PAPER NUMBER	
		2127		

DATE MAILED: 10/04/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	Application No.	Applicant(s)
	09/477,688	EDWARDS, STEPHEN ANTHONY
	Examiner	Art Unit
	Syed J Ali	2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### **Status**

1) Responsive to communication(s) filed on 07 May 2004.  
 2a) This action is FINAL.      2b) This action is non-final.  
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### **Disposition of Claims**

4) Claim(s) 1-12 is/are pending in the application.  
 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.  
 5) Claim(s) \_\_\_\_\_ is/are allowed.  
 6) Claim(s) 1-12 is/are rejected.  
 7) Claim(s) \_\_\_\_\_ is/are objected to.  
 8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### **Application Papers**

9) The specification is objected to by the Examiner.  
 10) The drawing(s) filed on \_\_\_\_\_ is/are: a) accepted or b) objected to by the Examiner.  
     Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
     Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).  
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### **Priority under 35 U.S.C. § 119**

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
 a) All    b) Some \* c) None of:  
 1. Certified copies of the priority documents have been received.  
 2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.  
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### **Attachment(s)**

1) Notice of References Cited (PTO-892)  
 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)  
 3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
     Paper No(s)/Mail Date May 7, 2004.

4) Interview Summary (PTO-413)  
     Paper No(s)/Mail Date. \_\_\_\_\_.  
 5) Notice of Informal Patent Application (PTO-152)  
 6) Other: \_\_\_\_\_.

## **DETAILED ACTION**

1. This office action is in response to the amendment filed on May 7, 2004. Claims 1-12 are presented for examination.
2. The text of those sections of Title 35, U.S. code not included in this office action can be found in a prior office action.

### ***Claim Rejections - 35 USC § 103***

3. **Claims 1-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lin (“Efficient Compilation of Process-Based Concurrent Programs without Run-Time Scheduling”) in view of Nilsen et al. (USPN 6,081,665) (hereinafter Nilsen).**
4. As per claim 1, Lin teaches the invention as claimed, including a method performed by a data processing system having a memory, comprising the steps of:
  - inputting a CCFG (pg. 213, Figs. 2[a-b], 3[a-b]);
  - scheduling the CCFG to produce a scheduled CCFG (pg. 215);
  - selecting a first node of the scheduled CCFG (pg. 213, Figs 2[a-c], elements p1, p2);
  - producing a first copy of the first node for an SCFG (pg. 213, Figs 2[a-c], elements p1, p2); and

coupling, if a first thread of the first node is suspended, between a second node of the SCFG of a second previously running thread and the first copy, a first context switch (Fig. 2[c], element c2).

5. Nilsen teaches the invention as claimed, including the following limitations not shown by Lin:

wherein the context switch saves a second state, of the second previously running thread, into a state variable dedicated to the second previously running thread (col. 37 lines 60-67).

6. It would have been obvious to one of ordinary skill in the art to combine Lin with Nilsen since the method of combining concurrent control flow graphs into sequential control flow graphs disclosed by Lin takes into account context switching between separate processes or threads, but fails to specify exactly how the code translation is generated, or what features are implemented therein. Rather, the preliminary ideas behind the construction of Petri net representations of sequential control flow graphs is disclosed, while the features related to scheduling and optimization are left up to the developer (pg. 215). The papers that Lin refers to presumably offer several scheduling techniques that can be used for the sequential control flow graph, yet the possible scheduling techniques are not limited to those references. Nilsen offers a technique for protecting the state of a thread during a context switch or a preemption by saving the information related to the suspended thread in a state variable and utilizing the information in the state variable upon resumption of the stopped thread, which would allow the thread and its shared resources to remain in a consistent state during any context switches or synchronization.

7. As per claim 2, Nilsen teaches the invention as claimed, including the method of claim 1, wherein the first context switch is comprised of code that saves a state of a thread being suspended in a state variable (col. 37 lines 60-67).

8. Lin teaches the invention as claimed, including resuming another thread by performing a multiway branch on a state variable for a thread being resumed (pg. 213, Figs. 2[c] and 3[c]).
9. As per claim 3, Lin teaches the invention as claimed, including the method of claim 1, wherein the translation of the CCFG into the SCFG produces, for each node of the CCFG, at most one corresponding node in the SCFG (pg. 213).
10. As per claim 4, Lin does not specifically disclose the method of claim 1, wherein the step of scheduling further comprises a topological sort for determining the scheduled augmented CCFG.
11. “Official Notice” is taken that the use of topological sorts is well known and expected in the art, and would have been an obvious modification to Lin. Specifically, Lin teaches an ordered graph, with nodes used to refer to specific variable states and edges to traverse between those states (pg. 212-213, Figs. 2[a-c], 3[a-c]). A topological sort is well known as a way of ordering nodes, such that if a transition occurs between two nodes, then based on how that transition is represented, it is known what order the nodes occur in the graph. A common way of expressing this is that if an edge exists such that edge(u, v) is in the graph and u and v are nodes in the graph, then u comes before v in the ordering of the graph. This can be found in any number of programming guides, and an example is presented in the Boost Graph Library on the Boost C++ Library website ([www.boost.org/libs/graph/doc/topological\\_sort.html](http://www.boost.org/libs/graph/doc/topological_sort.html)). It would have been obvious to one of ordinary skill in the art to use a topological sort to determine the ordering of the scheduled augmented CCFG since the technique is well known, other

programming methods that have been previously devised can be used in accordance with the sorting technique. Specifically, defining the graph with similar data structures would allow a programmer a multitude of predefined methods to operate on the data therein, depending on the specific needs of each individual system. Furthermore, Lin makes mention of how various features of the sequential control flow graph must be mapped out in order to generate the necessary code. This is done via a pre-ordering method, which represents the flow of control in various data structures, including a representation of all the “reachable nodes”. A topological sort would have been an obvious method of generating such a set of “reachable nodes” by defining all the nodes of the graph as well as the state transitions within a single data structure.

12. As per claim 5, Lin teaches the invention as claimed, including the method of claim 1, wherein an execution of the SCFG comprises translation of the SCFG into a programming language (pg. 213-216, §4.2).
13. As per claim 6, Lin teaches the invention as claimed, including the method of claim 5, wherein the programming language is C (pg. 213-216, §4.2).
14. As per claim 7, Lin teaches the invention as claimed, including the method of claim 1, further comprising a step of translation of the SCFG into a programming language (pg. 213-216, §4.2).

15. As per claim 8, Lin teaches the invention as claimed, including the method of claim 7, further comprising a step of executing the programming language translation of the SCFG (pg. 213-216, §4.2).

16. As per claim 9, Lin teaches the invention as claimed, including the method of claim 1, wherein an execution of the SCFG comprises interpretation of the SCFG (pg. 213-216, §4.2).

17. As per claim 10, Lin teaches the invention as claimed, including a data processing system having a memory and capable of implementing the method of claim 1 (Abstract).

18. As per claim 11, Lin teaches the invention as claimed, including a computer program product comprising a computer usable medium having computer readable code embodied thereon and capable of implementing the method of claim 1 (Abstract).

19. As per claim 12, Lin teaches the invention as claimed, including a computer data signal embodied in a carrier wave and representing sequences of instructions which, when executed by a processor, cause performance of the method of claim 1 (Abstract).

***Response to Arguments***

20. Applicant's arguments filed May 7, 2004 have been fully considered but they are not persuasive.

21. Applicant argues on page 3, “*The Examiner is factually incorrect when he states that Lin 02/1998 fails to specify exactly how the code translation is generated.*’ As Applicant has explained in his previous response, Lin 02/1998 teaches code generation according to a specific technique: *the use of a single program counter.*” Thus, Applicant alleges that the combination of Lin and Nilsen “*would change the principle of operation*’ of Lin 02/1998.”

Applicant adds that the publication submitted in the Information Disclosure Statement filed in the present response gives weight to Lin 02/1998 was not combinable with Nilsen, in that “*Lin 10/1999 differs...in its use of thread-dedicated program counters, [and t]he large time gap between the two papers, and the fact that the thread-dedicated technique was submitted for publication, are strong evidence that Bill Lin himself regarded the use of thread-dedicated program counters as a significant advance over his earlier work.*”

22. While Applicant’s assertion that the Lin reference relied upon in the rejection uses a program counter to trace the synthesized program’s execution has been noted, Examiner respectfully disagrees. Whether or not Lin uses such a program counter does not preclude the fact that Lin teaches a method of taking multiple concurrent processes and synthesizing those processes into a single sequential program by joining the multiple processes at common transitions where a context switch may occur. This idea is broad enough in nature to have applications in many types of environments and programming languages. While Lin teaches that this translation is based in a language developed with extensions to C, the concepts are applicable to other programming languages and environments, and should be viewed so narrowly as to eliminate these additional applications. While Lin does not specifically address thread based execution and context preservation, such a model would clearly stand to benefit from the

synthesis model taught by Lin. Multithreading makes extensive use of concurrent processes, and any means of eliminating excessive context switches would be beneficial. As Lin leaves the specific scheduling heuristic up to the developer, it stands to reason that the developer also could use whatever programming paradigms necessary to achieve the appropriate goals. Nilsen provides a method of scheduling threads in a virtual machine, wherein Java is a prevalent form of software development, and would stand to benefit greatly from a method of reducing context switches and the overhead incurred with those context switches.

*Conclusion*

23. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (571) 272-3769. The examiner can normally be reached on Mon-Fri 8-5:30, 2nd Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai T An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

  
Syed Ali  
September 29, 2004

  
MENG-AI T. AN  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100